

Project Proposal

Build a coverage adapter on top of the new Jenkins [code-coverage-api-plugin](#) for your favorite Electronic Design Automation (EDA) tool coverage report. It may be either a tool widely used in the industry (VCS, Questa Sim, Quartus, etc.) or an open-source tool (e.g. [covered](#)). The end goal is to get EDA coverage reports published in Jenkins, including summary reports, trends, and unified APIs.

The types of coverage reports are related to the types of EDA tools. There are two classes of EDA tools, and they produce very different types of information, but they can all be considered “coverage”:

1. Synthesis and Backend tools
2. Compilation and Simulation tools

Types of coverage reports that could be supported are:

- Resource utilisation levels:
 - LUT usage level
 - Clock tree usage level
 - PLL usage level
 - Routing resources usage level
 - Total device utilization
- Timing information
 - minimum slack
 - device maximum speed
 - timing margin
 - best case worst case timing
 - path delays
 - others
- VHDL/Verilog/SystemVerilog Code coverage
 - Line coverage
 - Branch coverage
 - FSM coverage
 - Verilator specific, see the [verilator manual](#):
 - underscore coverage
 - user coverage
 - others
- Simulation related coverage
 - assertion coverage
 - functional coverage
 - UVM message coverage (number of message for each type)
 - Test coverage
 - others
- Test
 - test ranking (which tests contribute to the most/least coverage)
 - others

These are ideas of items that could be covered in the project proposals. It is obviously impossible to do them all in one plugin, as they are specific to each EDA tool. A good approach would be to focus on a single EDA tool, and build the coverage adapter for what that EDA tool reports.

Newbie-friendly issues

- Code Coverage API: [JIRA Query](#)

Quick-start Guides

- Explore articles about coverage analysis in Electronic Design Automation, e.g. [Verilog code coverage with covered and iverilog](#)
- Select an EDA tool you would be interested in integrating with Jenkins. Your personal experience with a tool would be a huge plus for the project, but you can also take any other tool you are interested in
- Review the suggested coverage use-cases above, build a list for the selected tool
- Explore ways to generate the desired coverage reports with the selected EDA tool, e.g. from its CLI. If it is possible to produce a machine-readable format, it should be possible to integrate it with Jenkins